REMARKS

The above amendment is made in response to the Final Office Action mailed July 18, 2004. Claims 1-34 are pending in the present application. Claims 26-34 have been allowed. Claims 7, 8, 10-15 and 22 are objected to. Claims 1-6, 9, 16-21 and 23-25 stand rejected. The Examiner's reconsideration is respectfully requested in view of the following remarks.

Claims 1-6, 9, 16-21 and 23-25 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Agrawal et al. (U.S. Patent No. 5,799,311) (hereinafter "Agrawal") in view of Ramaswamy et al. (hereinafter "Ramaswamy"). The rejection is respectfully traversed.

In Applicants' Amendment (Paper No. 4) to the first Office Action (Paper No. 3), Applicants argued that col. 3, lines 40-43 of Agrawal does not teach or suggest "constructing a decision tree from the input set...based upon *multivariate subspace splitting criteria*," as claimed in claim 1.

Citing col. 3, lines 40-43 of Agrawal, the Examiner states:

> as a means for creating a decision tree is created [sic] by repeatedly splitting the records at each examined node starting with the root node, at any examined node a split test is determined to best separate the records at that node by record class and using the attribute lists, the node's records are split according to the best split test into partitions of records to form child nodes of the examined node

Applicants reiterate that the "split test" disclosed in Agrawal creates *univariate* decision trees, and not *multivariate* decision trees, as created in claim 1. The decision tree construction algorithm in Agrawal represents a most primitive method. To better explain, it is important to understand that a *univariate* split involves using a *single* attribute to split each node. On the other hand, a *multivariate* split uses *multiple* attributes; thus, each split is capable of creating a "*multivariate subspace*," as claimed in claim 1. This multivariate

subspace *cannot* be created using univariate splits. It is respectfully noted that, in view of the disclosure, one skilled in the art would understand the difference between the application of a univariate and a multivariate splits.

The Examiner apparently argues that by repeatedly splitting the records from the root node to the leaf node, and by combining all univariate splits on the path, Agrawal creates a multivariate subspace. This reasoning is clearly flawed. Consider the following example, wherein "o" and "*" are objects of two different classes.

oooo***
ooo***o
oo***oo
o***ooo
***oooo

Using the method of Agrawal, one can choose only one attribute to split. Thus, one can make only one of two splits -- *horizontally* or *vertically*. Clearly, a horizontal or vertical split will not result in a pure partition. Hence, it requires many splits until each leaf node becomes pure. This results in many leaf nodes containing very small number of objects, thus failing to give one a good picture of how the objects are distributed.

On the other hand, multivariate splits allow one to split *diagonally* (or even *spherically*), instead of just horizontally or vertically. One can never find good splits in a spherical boundary using the univariate splits described in Agrawal. Using multivariate splits results in a much smaller number of leaf nodes, where objects of the same class are clustered. Thus, multivariate splits, as described in claim 1, are clearly distinguishable from multiple univariate splits. A benefit of multivariate splits is that it provides a good picture of how objects of different classes are distributed. Further, runtime classification cost is much lower (*e.g.*, univariate trees test one feature at a node and thus result in larger trees than if multiple features are tested at a node, as with multivariate trees). Thus, the

presently claimed invention provides clear advantages over the combination of Agrawal and Ramaswamy.

In the Final Office Action, the Examiner further argues that "Agrawal discloses the use of 'a nearest neighbor set of nodes for each of the leaf nodes based upon a respective closeness of the nearest neighbor set of nodes to a target record of the target class,'" as claimed in claim 1. The Examiner cites col. 6, lines 52-54 of Agrawal for support. Once again, the citation does not provide a valid foundation for the rejection

Col. 6, lines 52-54 of Agrawal states: "The records at each new leaf node are checked at block 23 to see if they are of the same class." Block 23 in Figure 7 of Agrawal refers to a step during the process of building a *univariate* tree from training data. More specifically, for each newly created node, block 23 checks whether a node is pure (*i.e.*, whether the records are of the same class) (Specification, p. 16, lines 4-5). If it is, this node is a leaf node; otherwise, an attribute is selected to further split the node. Agrawal effectively tries to find the *best attribute* for each node in order to split that node.

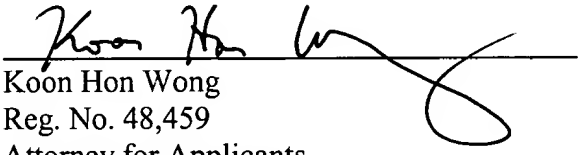
The concept of "a nearest neighbor set of nodes," as claimed in claim 1, addresses an entirely different issue from Agrawal. For each leaf node, we ask: "if a test object (*i.e.*, the example of which one wants to predict its class label) falls into this node, what are the other nodes it can possibly fall into?" In light of the specification, the present method finds for each leaf node, a set of possible nodes that this test object may fall into, and this set does not depend on the test object. Thus, we greatly reduce the number of nodes one needs to check. We call this set of nodes the nearest set for those leaf nodes, as essentially claimed in claim 1. It should be noted that Agrawal is totally irrelevant and inapplicable here, because with *univariate* splitting, each object falls into one node and one node only.

It is further submitted that the combination of Agrawal and Ramaswamy, as described by the Examiner, is not possible. Agrawal provides no mechanism of using distance functions. Test objects are classified into leaf nodes by following the path based on their attribute values. The splits described in Agrawal are along only one attribute and do not create arbitrary subspace. Thus, a distance function (*e.g.*, distance computed between the *center of the subspace* to the test object) is entirely unnecessary.

Accordingly, claim 1 is believed to be patentably distinguishable over Agrawal in view of Ramaswamy. Independent claim 24 and dependent claims 2-23 are believed to be allowable for at least the reasons given for claim 1. Dependent claim 25 is believed to be allowable for at least the reasons given for claim 24. Withdrawal of the rejection of claims 1-6, 9, 16-21 and 23-25 is respectfully requested.

In view of the foregoing remarks, it is respectfully submitted that all the claims now pending in the application are in condition for allowance. Early and favorable reconsideration is respectfully requested.

Respectfully submitted,

By: 

Koon Hon Wong
Reg. No. 48,459
Attorney for Applicants

F. CHAU & ASSOCIATES, LLC
130 Woodbury Road
Woodbury, NY 11797
Telephone: (516) 692-8888
Facsimile: (516) 692-8889